AMENDMENTS TO THE SPECIFICATION:

Please replace paragraph [0010] with the following amended paragraph:

[0010] DM Tools (e.g. SAS Enterprise Miner™ and ANGOSS KnowledgeSTUDIO™) are software applications installed on a dedicated computer system for use by skilled analysts to perform data mining tasks (i.e. data exploration or knowledge discovery and predictive modeling). These DM tools enable users to import, analyze and model data sourced from a variety of files, database tables, or other data structures. RDBMS Systems ~~(e.g. Microsoft SQL Server 2000 and Oracle 9i)~~ (e.g. Microsoft SQL Server 2000™ and Oracle 9i™) incorporate data mining algorithms into the relational database environment, enabling the mining discovery of or relationships between data stored in tables contained in the relational database.

Please replace paragraph [0011] with the following amended paragraph:

[0011] Thus a limitation of these systems is that rather then providing analyst interfaces, they typically provide an application programming interface that enables third parties to design and develop applications and interfaces integrated with these RDBMS Systems. DM Applications provide user interfaces and functionality enabling the application of data mining algorithms to data captured or accessible to the DM Application in a specific business domain. The data mining capabilities of DM Applications may be developed by the application provider ~~(e.g. E.Piphany, Blue Martini, etc.)~~ (e.g. E.Piphany™, Blue Martini™, etc.) , or through integration of the enterprise software application with a data mining engine incorporated into an RDBMS System, or using an integrated data mining engine that provides an application programming interface for this purpose (e.g. ANGOSS KnowledgeSERVER™ and KnowledgeSTUDIO™ SDK).

Please replace paragraph [0045] with the following amended paragraph:

[0045] In the following description, numerous specific details are set forth to provide a thorough understanding of the invention. However, it is understood that the invention may be practiced without these specific details. In other instances, well-known software, structures and techniques have not been described or shown in detail in order not to obscure the invention. In addition, the system and method may be implemented using a number of different programming languages ~~(e.g. Java and Enterprise Java Beans)~~ (e.g. Java™ and Enterprise Java Beans™) and for a variety of different platforms ~~(e.g. UNIX and Linux)~~ (e.g. UNIX™ and Linux™). The following description illustrates the invention by reference to the ~~Microsoft~~ Microsoft™ platform comprising the ~~Windows~~ Windows™ operating system, ~~SQL Server~~ SQL Server™ relational database server, and Internet Information ~~Services~~ Services™ web server, but other platforms, database server, web servers, applications and programming tools may also be used. In the drawings, like numerals refer to like structures or processes.

Please replace paragraph [0050] with the following amended paragraph:

[0050] The term "data mining provider" is used herein to refer to an application server (e.g. ANGOSS KnowledgeSERVER™, Microsoft SQL Server Analysis ~~Services~~ Services™, or an implementation of an application server under JSR 073, a ~~Java~~ Java™ Community standard) that may train data mining models (i.e. based on the analysis of data) and call existing data mining models to generate scores (i.e. based on a new data instance).

Please replace paragraph [0055] with the following amended paragraph:

[0055] The invention, may be deployed within enterprise software applications, including, for example, the Siebel Systems, Inc.'s ~~("Siebel")~~ ("Siebel™") suite of "eBusiness Applications" ~~("eBusiness")~~ ("eBusiness™") . The ~~Siebel eBusiness~~ Siebel eBusiness™ application suite is a leading "customer relationship management" ("CRM") software application and is representative of standard CRM environments. The invention supports the

creation of data mining models using tools and interfaces designed specifically by reference to the ~~Siebel~~ Siebel™ data schema, and the deployment of data mining models and associated presentations to ~~Siebel eBusiness~~ Siebel eBusiness™ applications in several modes including applets working within the interface environment and "pop up" windows working within the ~~Siebel eBusiness~~ Siebel eBusiness™ environment. Through integration with the ~~Siebel eBusiness~~ Siebel eBusiness™ application infrastructure, the invention's analytic decision engine treats all ~~Siebel eBusiness~~ Siebel eBusiness™ interfaces ~~(e.g. Siebel "eSales", "eService", "eCallCenter", "ePersonalization", and "ePricer")~~ (e.g. Siebel™ "eSales™", "eService™", "eCallCenter™", "ePersonalization™", and "ePricer™") as target environments for the delivery of presentations based on the deployment of data mining models. Integration options may include using ~~Siebel~~ Siebel™ applets working within the ~~Siebel~~ Siebel™ interfaces themselves, or with browser based messaging from the invention's analytic decision engine to the target environment, or using the system itself as an integrated application within the ~~Siebel eBusiness~~ Siebel eBusiness™ environment.

Please replace paragraph [0057] with the following amended paragraph:

[0057] *Data Mining System Functional Components*. Referring to FIG. 2, there is shown a block diagram of a data mining system **200** in accordance with an embodiment of the invention. The data mining system **200** includes the following: an analytic decision engine ("ADE") **210**; a model repository **220** for the storage of scores, rules, and rules sets resultant from multiple data mining tasks; a data mining management console **230**; and, means for importing into the repository **220** a wide variety of data mining models **240** and scores **250** generated from data mining processes involving diverse DM Tools ~~(e.g. SAS Enterprise Miner, IBM Intelligent Miner, ANGOSS KnowledgeSTUDIO)~~ (e.g. SAS Enterprise Miner™, IBM Intelligent Miner™, ANGOSS KnowledgeSTUDIO™) , RDBMS Systems ~~(e.g. Microsoft SQL Server 2000)~~ (e.g. Microsoft SQL Server 2000™) , and DM Applications.

Please replace paragraph [0060] with the following amended paragraph:

[0060] In general, the analytic decision engine **210** includes means designed to accelerate the data mining process and provide a flexible, extensible, and highly scalable environment for deploying and managing a number of different data mining models and associated presentations. It may support, for example, the following: (a) the importing of data mining models **240** and scores **250** together with conversion utilities for converting proprietary data mining models (e.g. ~~SAS~~ SAS™ models) into XML documents to support applications of the analytic decision engine **210** on a vendor, platform, and system neutral basis; (b) a model repository **220** for the storage of these data mining models **240**, as well as for optimization of these models and supporting other features of the system; (c) a data mining management console **230** which provides means to enable a human user to manage, evaluate, and optimize a plurality of data mining models **240** operating in one or more digital ecosystems; and, (d) an application server **260** which supports the deployment of data mining models to users and interfaces connected to the application server directly by wired or wireless connections **290**, **291**, **292** or through an application, interface, or appliance connected to the network by wired or wireless connections **270**, **280**.

Please replace paragraph [0062] with the following amended paragraph:

[0062] *Scoring Services and Presentation Services*. Referring to FIG. 3, there is shown a block diagram illustrating typical interfaces **300** for the data mining system **200** in accordance with an embodiment of the invention. The primary modules within the ADE **210** include presentation services **211** and scoring services **212**. The ADE **210**, presentation services **211**, and scoring services **212** integrate with the typical interfaces **300** which may include the following: a browser **310**, desktop applications **320**, both of which may communicate with ~~Windows 2000/IIS~~ Windows 2000/IIS™ (internet information server) **340** via HTTP **330**; COM+(component object model extension) **350** links between ~~Windows 2000/IIS~~ Windows 2000/IIS™ **340** and the analytic decision engine **210**; OLEDB-DM (application program interface) links between the analytic decision engine **210** and OLEDB-DM servers **390**; and, HTML, XML, COM (component object model), DCOM (distributed

component object model), or SOAP (simple object access protocol) **360** links between the analytic decision engine **210** and other servers and clients **260**.

Please replace paragraph [0064] with the following amended paragraph:

[0064] The MPS module **211** includes the following functionality:

- Generation of HTML to display in a presentation environment of a client-server or browser server based application such as on a web site or to a customer service representative (i.e. in the manner which will be described below with reference to FIGS. 4 through 7).

- Monitoring the use of data mining models by providing an audit file (i.e. as part of the repository **220** a representative schema of which is illustrated in FIG. 8) which records relevant data with respect to the use of simple and complex models with related means to define parameters for such recording actions (e.g. frequency) by reference to specific models and other measures. This functionality enables a user to manage a wide range of simple and complex models operating in a variety of different systems and to assess the performance of these models.

- The incorporation of appropriate content into the HTML being presented that, in conjunction with server modules, records the outcome of the running of the models and presentations in the repository **220**, as illustrated in FIG. 8, for use in training new models, in optimizing existing models, and in modifying or replacing presentations associated with existing models.

- The MPS may also return the presentation ID list as opposed to the HTML which conveys those presentations. This capability allows for additional features described below.

- The ability to benchmark performance of MPS **211** and SS **212** by applying a long list of customers in sequence to a presentation definition. This may be accomplished by running a data batch processing routine that performs the same operations done in a real time mode. Performance statistics may be gathered and analyzed by running the presentation definition in this batch mode to estimate performance and hardware requirements for a real time system.

- The ability to generate overview reports that show the overall distribution of the presentations. A presentation editor may allow users to create a complex series of rules and formulas that determine which presentation to give to a particular individual. As such, it is easy to loose site of the over all effect of these. By applying the presentations to a complete list or statistically representative sample of customers and tabulating the resultant presentations, a user may gain an overall view of the presentations.

- The ability to export a list of presentations that may be imported by a marketing automation application. This allows an organization to have a unified presentation strategy across all marketing channels.

- In order to achieve high scalability, an instantiated MPS object may be heap-lock free (i.e. runtime read only), multithreaded, and initialization stateless. Instantiating and initializing an MPS object is time consuming. It may digest all the presentation rules and formulae, pre-compile any SQL data acquisition or scoring commands, and load the presentation HTML into memory. As such, it should not be initialized during each score request. Instead, the object should be instantiated once when it is first used or at another appropriate time. This means that the object may exist on the server for an extended period of time (e.g. stored in the ~~Windows~~ Windows™ Running Object Table or ~~ASP~~ ASP™ Application object) and many different users (i.e. execution threads) may be simultaneously using the MPS object. In order for this not to cause performance bottlenecks, the MPS object should not require many (any) mutex locks required when writing to heap memory.

- Because MPS is designed for operational environments and live objects may be retained in memory, updates to presentation definitions may implement special handling. In order to handle this situation, presentation objects have a version number (refer to the "ASYS_Projects" table **810** contained in FIG. 8) and the moniker (name) of the object has the ID and version number. When a new version of the object is saved, the version number is incremented. The next time the presentation object is requested the new most up-to-date one will be instantiated and initialized. If any old objects are still in use, they will run as normal.

- When a calling module (e.g. client browser or another server) calls MPS to request the results of a presentation, a "Model Locator" may be used to refer to the presentation. This locator in turn links to one or more presentations. This allows a presentation to be changed without having to modify any clients. For example, all clients could use the URL query string "PresentationLocator=StdCC&EntityID=12345". In this case, "StdCC" is the ID of the locator. This locator ID is looked up to discover the actual presentation. In order to have clean and unbiased data on the benefits of one presentation versus another, the locator ID may actually link to more than one presentation. One of the presentations may be a test group. Some exposures will be given to the test presentation on a random or pre-selected customer list basis. This allows for valid comparisons between two or more presentations.

- The client may also pass data to the ADE **210** for use by dynamic models. The query string "PresentationLocator=StdCC&EntityID=12355&Age=25&Sex=M" is an example of this. In this example, if any of the dynamic models require inputs "Age" or "Sex" then the passed values will be used, even if a data lookup can also fulfill the data request. Furthermore, the query string, "PresentationLocator=StdCC&Age=25&Sex=M" does not include an entity ID. In this case, all the required data must be provided and all the models must be dynamic. This feature allows for "anonymous CRM" (i.e. client relationship management).

Please replace paragraph [0071] with the following amended paragraph:

[0071] MPS **211** also provides for the ordering of different presentations (e.g. through tagging and sorting presentations based on model scores as "Beginning", "Sorted", or ~~"End".)~~ "End"). As a result, the analytic decision engine **210** provides for the following:

- the application, via the scoring services API **212** within the analytic decision engine **210**, of predictive models to transactional systems, for example through the reading and analysis of transactions from a message queue service, such as IBM's ~~MQSeries~~ MQSeries™ or Microsoft's Message Queuing Service ~~(MS MQ)~~ (MS MQ)™, enabling the identification of predicted outcomes (e.g. fraud, defection, best bid or other valuable knowledge relative to business decision-making);

- the exposing of these scores for interrogation by MPS **211**; and,

- the resultant serving of instructions to MPS **211** which proceed to serve appropriate HTML in the form of messages, lists, suggestions or any other defined HTML content wished to be presented to support a transaction, interaction or other action based on the scores defined and determined within SS **212** enabling the triggering of appropriate actions (e.g. routing transactions determined to have a high fraud score via SS **212** to investigators through MPS **211**).

Please replace paragraph [0073] with the following amended paragraph:

[0073] By enabling MPS **211** and SS **212** to be accessed with HTTP, COM, SOAP, and XML as a representative integration platform, the system allows the analytic decision engine **210** to service a wide variety of clients and servers. The analytic decision engine **210** is based on industry standards (e.g. ~~Microsoft~~ Microsoft™ and Internet technologies, including Windows ~~2000~~ 2000™ and COM+MTS) which make the analytic decision engine

210 highly scalable. Access to the repository **220** may be accomplished via a programmatic interface that enforces accessibility rules.

Please replace paragraph [0078] with the following amended paragraph:

[0078] These functions make use of industry standards (e.g. XML) to interoperate with generally available browser solutions (e.g. Internet ~~Explorer~~ Explorer™). Documents incorporated in the data mining system may be formatted based on XML standards and served in this format. Optimizations and transformations may be implemented with these standards. Since transformations are performed in the browser, performance is improved.

Please replace paragraph [0085] with the following amended paragraph:

[0085] Referring to FIG. 5(a), there is shown a block diagram illustrating the integration **500a** of a traditional client-server or "thick" desktop application with the analytic decision engine **210** in accordance with an embodiment of the invention. This integration may be accomplished through the use of a browser ~~Active X~~ ActiveX™ control embedded in the application which is configured to determine (i.e. through the data mining system **200**) and display a presentation. Referring to FIGS. 2, 3, and 5(a), desktop applications **320** running on personal computers may gain access to the analytic decision engine **210** using a ~~Browser~~ browser ~~Active X~~ ActiveX™ control that is included as a standard component of ~~Microsoft~~ Microsoft™ Internet ~~Explorer~~ Explorer™ (i.e. version 4 or 5) ("IE"). For non-IE environments, the analytic decision engine **210** may be configured to support the ~~Mozilla~~ Mozilla™ browser which is an open source implementation of the ~~Netscape~~ Netscape™ browser. These controls may be distributed to users who do not have or want IE. Once the ~~Browser~~ browser ~~Active X~~ ActiveX™ control is placed in a desktop application **320**, a user may communicate with the presentation services **211** by navigating to the correct URL (uniform resource locator). As such, the same solution may service both thin and desktop applications.

Please replace paragraph [0087] with the following amended paragraph:

[0087] Referring to FIG. 5(b), there is shown a block diagram illustrating an interface suggestion applet **500b** in accordance with an embodiment of the invention. This is an example of desktop application integration, as illustrated in FIG. 5(a), with reference to the ~~Siebel~~ Siebel™ "Call Center" application which is a traditional client-server application developed by a third party software developer.

Please replace paragraph [0088] with the following amended paragraph:

[0088] Referring to FIG. 5(c), there is shown a screen capture of an interface suggestion applet **500c** in accordance with an embodiment of the invention. This is an example drawn from the ~~Siebel~~ Siebel™ Call Center application as illustrated in FIG. 5(b). The screen capture **500c** illustrates the analytic decision engine's **210** suggestion applet **500b** running in a standard "customer relationship management" software application, namely, the ~~Siebel~~ Siebel™ "Call Center" environment. In this example, a customer service representative user ("CSR") is provided with dynamic suggestions with respect to the best offer to make to the customer, who the CSR may be currently interacting with, based on the use of the analytic decision engine **210** to deliver data mining model outputs and associated presentations (in this case a customer offer) to the CSR. The analytic decision engine **210** supports this form of integration with a wide variety of different enterprise applications developed using industry standard development tools and programming languages ~~(e.g. Microsoft Visual C++, Visual STUDIO, Visual Basic, Java, etc.)~~ (e.g. Microsoft™ Visual C++™, Visual STUDIO™, Visual Basic™, Java™, etc.) .

Please replace paragraph [0089] with the following amended paragraph:

[0089] The analytic decision engine **210** may be integrated with software applications in a variety of ways, taking into account the method of development and deployment of the

broad range of different client-server ("thick client") and web-server ("thin client") software solutions. Two of these methods, which may be illustrated with reference to the ~~Siebel~~ Siebel™ enterprise application platform, are via suggestion applet or pop-up window. With respect to suggestion applets, and referring to FIGS. 5(b) and 5 (c), a user may employ ~~Siebel~~ Siebel™ tools to create suggestion applet **500b** calls to the analytic decision engine **210**. This suggestion applet **500b** may contain the user's choice of browser ~~Active X~~ ActiveX™ control and may call the presentation services **211**, as described above, with additional template code. Using this approach, suggestions appear as part of the ~~Siebel~~ Siebel™ application. This is advantageous when data mining forms part of development specifications or when major changes are being made to the ~~Siebel~~ Siebel™ application.

Please replace paragraph [0091] with the following amended paragraph:

[0091] In other words, and referring to FIGS. 6(a) and 6(b), the suggestion window may be a pop-up window **600a**. In this case, the application monitor **610b** shown in FIG. 6(b) polls the Siebel application every second or two. When the user enters an area that warrants a suggestion, a window pops up. Where and when the window is popped up is managed via the data mining management console **230** as described above. This approach has the advantage that new suggestions may be added to various locations without ~~Siebel~~ Siebel™ repository changes (i.e. except, of course, for a simple initial change). This embodiment is advantageous for adding data mining capabilities to existing applications and for deploying predictive intelligence out to business users "on their terms" and through the interfaces that they are comfortable with. The embodiment has the following additional advantages: predictive models and related score defined presentations may be deployed across all ~~Siebel~~ Siebel™ applications in a common and consistent manner; there is no disruption to the ~~Siebel~~ Siebel™ data model or operational environment; and, outcomes may be moved directly to the analytic data mart for analysis, iteration, and optimization. The embodiment also includes the following advantageous features: platform independent; database independent; data mining and modeling tool independent; easy to implement and rapidly deploy; easy to manage and monitor at both business and technical levels; and, easy to

implement from a training perspective because front office customer facing personnel and customer touching CSRs are presented (i.e. through their browsers or other appliances) with content and outcomes only. Moreover, the analytic decision engine **210** remains CRM platform and CRM interface agnostic. The analytic decision engine **210** may support deployment of predictive intelligence to legacy environments, other CRM application platforms, and point solutions designed to fulfill defined requirements of an organization's CRM business and technology plans not addressed by ~~Siebel~~ Siebel™ or other similar third party CRM application vendors.

Please replace paragraph [0094] with the following amended paragraph:

[0094] *Data Mining Repository and Data Mining Management Console*. Again referring to FIG. 8, the data mining repository **220** stores data relating to the analytic decision engine's **210** scoring services **212** and presentation **211** modules. This repository **220** includes the following:

- Simple and complex "Scores"; "Presentations" (or instructions for locating such Presentations), including related presentation rules, which may be any number of different HTML based forms of documentation, links, instructions, images or other content determined to be or desired to be associated with scores contained in the model repository; "Constraints" being any one or more user defined or system defined rules, rules system or constraints which supplement data mining defined predictive model scores, rules and rules sets with other user defined or system defined rules and rules sets to manipulate the manner in which Presentations based on Scores are used or not used; and, "Outcomes", which may be numeric or text based and which are associated with the serving of one or more Presentations through time as described above for purposes of assessing and monitoring the performance of the system;

- Conversion utilities for converting proprietary data mining models (e.g. ~~SAS~~ SAS™ models) operating in production environments into XML documents to support

applications consuming predictive models on a vendor, platform, and system neutral basis;

- Support for the use of any combination of static or dynamic models or scores, together with composites thereof and other user defined constraints and rules as inputs in the analytic decision engine rules engine;

- Support for the integration of scores with user defined presentations to be served in connection with any person to person, person to system, or system to system interaction in the form of HTML, together with related optimization capabilities enabling the monitoring of results from the deployment of such models, scores and/or presentations to enable the continuous improvement of the capabilities of the system; and,

- An architecture suitable for a wide range of computing platforms and based on highly scalable industry standard technologies to enable the convenient implementation of the invention (i.e. availability of COM and SOAP interfaces for cross-platform implementation; use of COM+MTS, HTTP, and XML for high scalability; client side ~~JavaScript~~ JavaScript™ libraries to call the scoring server for dynamic adjustment to HTML forms and content based on model results).